Portable DMX Tester

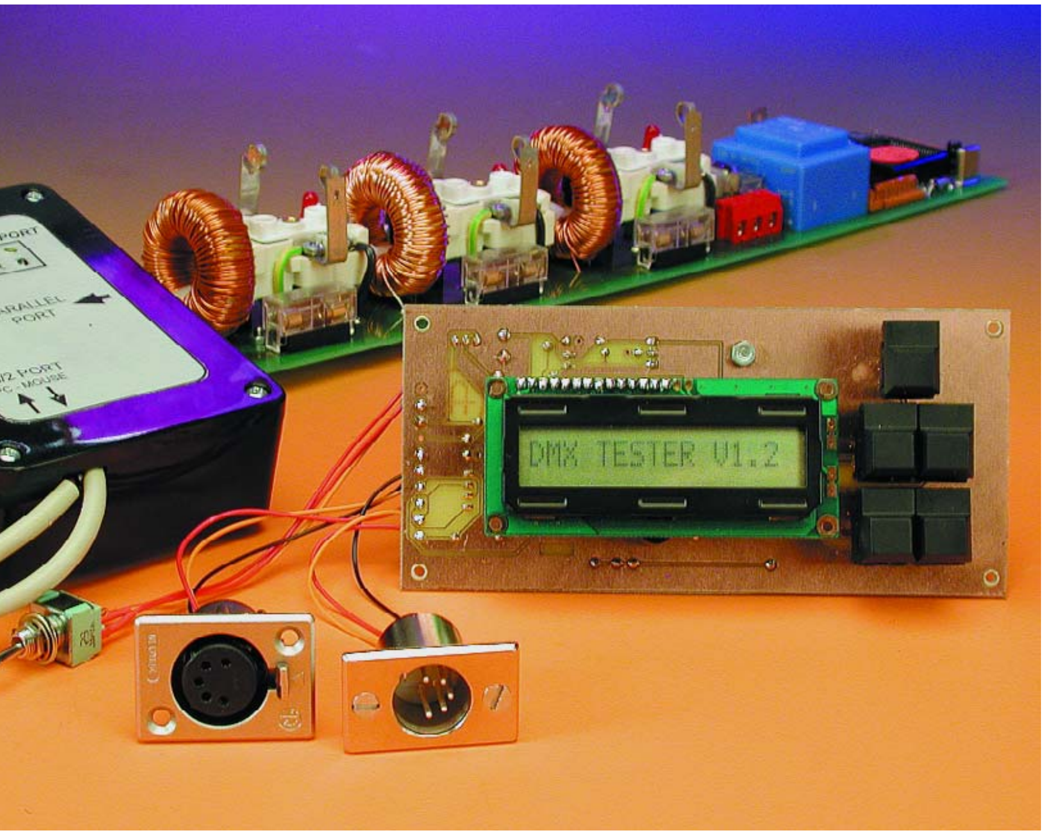
everything under control

Design by B. Bouchez

It is frequently necessary that a DMX-compatible device or a complete DMX installation has to be tested when faults have occurred while it is in use. With the tester described here we avoid the need to have to take over the DMX master (PC or console) for this purpose. This tester can be used in every situation, thanks to its handy size and built-in rechargeable battery. And as an added benefit, this circuit can also send and receive DMX data.

Anyone who has ever installed or built DMX-compatible equipment has, in all likelihood, experienced problems when it came to the operation thereof.

It is impossible to describe the wide variety of possible faults and their symptoms here, because of the enormous scale of the subject! Depending on circumstances, it is possible that a simple piece of equipment (scanner, dimmer, etc.) doesn't respond to a command, or that a complete installation has suddenly gone haywire and is carrying out programs that only it seems to be familiar with…



When the installation consists of a simple 0 to 10-V control system, a simple multimeter is usually sufficient to trace the problem and work out a solution.

When DMX512 is concerned however, fault finding with the aid of only a multimeter or oscilloscope is equal to searching for a needle in a haystack. When seriously tackling a fault in DMX512 equipment or an entire installation, a tool is necessary that can send valid commands (according to the DMX512 standard) and also interpret and display whatever is happening at any given instant along these connections.

It is certainly possible to use a DMX interface that is connected to a PC, but this solution requires a considerable investment in a device that is used only for testing purposes. In addition, it is not very practical to carry a PC (including a portable) to a height of several meters, the location of the equipment to be tested.

The tester proposed here provides the facility to solve various problems in a very simple manner. It is sufficient to glance at the specifications, listed in **Table 1**, to be convinced of this.

The Schematic

Only a few years ago, the building of a tester with all these features would have required a circuit with considerable complexity. But thanks to the use of a modern microcontroller, practically everything that is required fits on a single chip. Therefore, the schematic shown in **Figure 1** appears to be simplicity itself. The trade-off however, is that a significant amount of effort has to be invested in the software.

All the work is carried out by IC1, an Atmel microcontroller from the AVR-family. It has everything we require on board: 8 kBytes of flash-memory for the program, 512 bytes of RAM, two timers, a serial interface and even 512 bytes of EEPROM (which is not actually used here).

The whole system is clocked by an 8-MHz crystal, which results in a throughput of 8 MIPS (MIPS = Million Instructions Per Second), plenty of computing power for the present application.

IC1 is an AT90S8515 and it even has its own programmer on board. The only things we need in order to load our program into flash memory, are an appropriate cable that connects to the parallel port of a PC and a program that Atmel has made available free of charge. This cable is connected to K2, which need not be mounted if you decide to purchase a pre-programmed controller from *Elektor Electronics*.

The reset circuit around transistor T1 appears a little strange at first. However, this makes it possible to not only reset the circuit during normal use with the combination of R2/C3, but also during the programming stages of the Atmel program.

The interface for the LCD has been implemented as straightforward as possible. Port A of IC1 is configured for bi-directional operation and takes care of the exchange of data. The control signals are managed by lines on ports C and D. The program ensures the correct timing for the LCD controller.

The five control buttons are connected directly to the inputs of port B.

The interconnection with the DMX512 world is realised with a chip that will be familiar to many

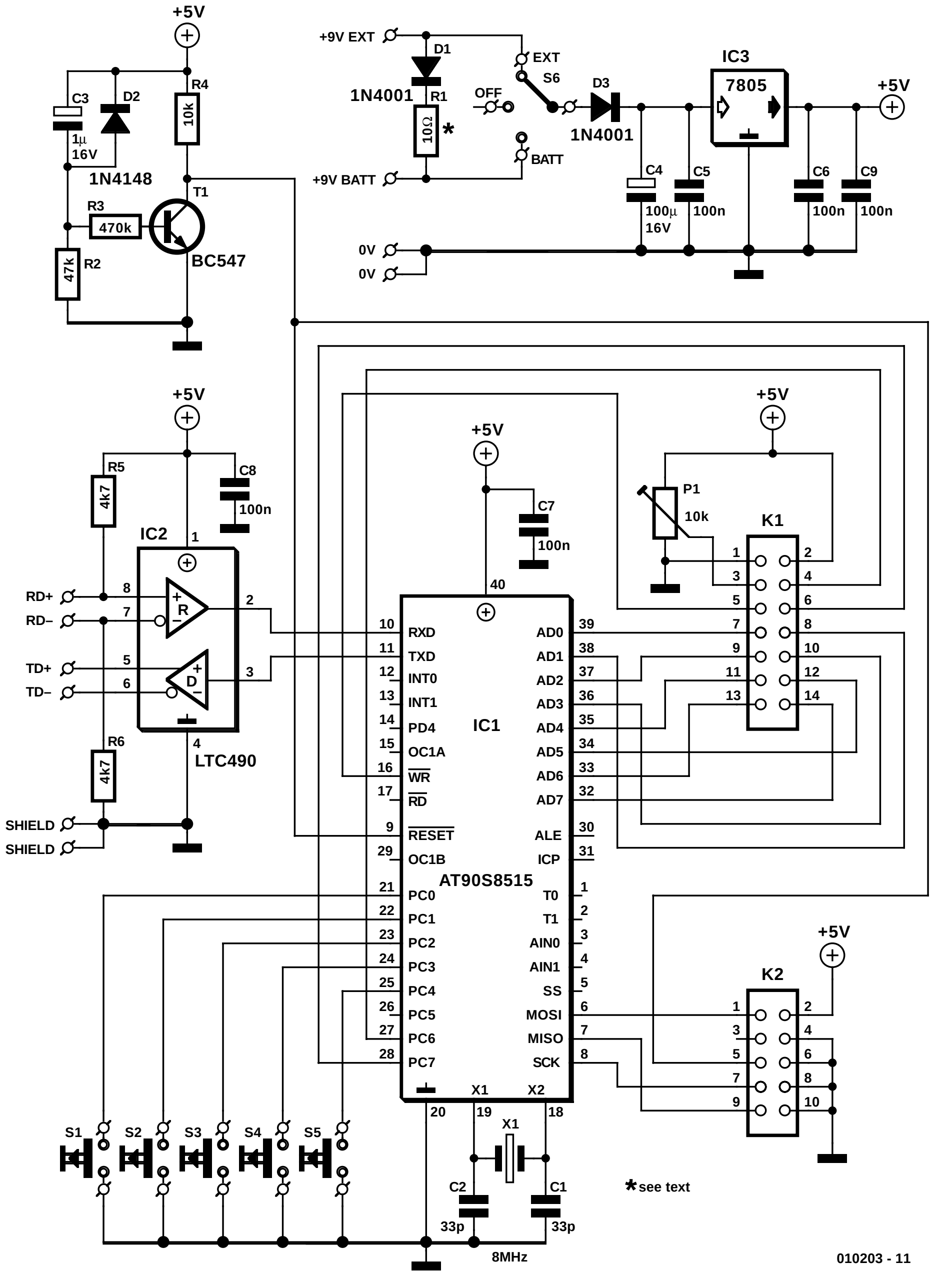


Figure 1. The schematic for the portable DMX tester is not very complicated, despite the fact that this is a complete microcontroller system.

Table 1. Specifications for the Portable DMX Tester

- – Portable tester with modest dimensions
- – Compatible with DMX512-1990
- – Transmitted and received data visible on an LCD
- – Very straightforward operation (5 buttons)
- – Operates from rechargeable battery or mains adapter
- – Two functions: DMX transmit and receive
- – Transmits on 512 DMX channels (480 are adjustable, the others remain 0)
- – Start Code during transmission can be adjusted as desired
- – Receipt of 480 channels, the values of which are indicated from 0 to 255
- – Indication of the received Start Code
- – Indication of received data, so that a break in the cable can be detected immediately
- – The option of storing the received DMX-values in the memory of the tester and re-transmit them again at a later time

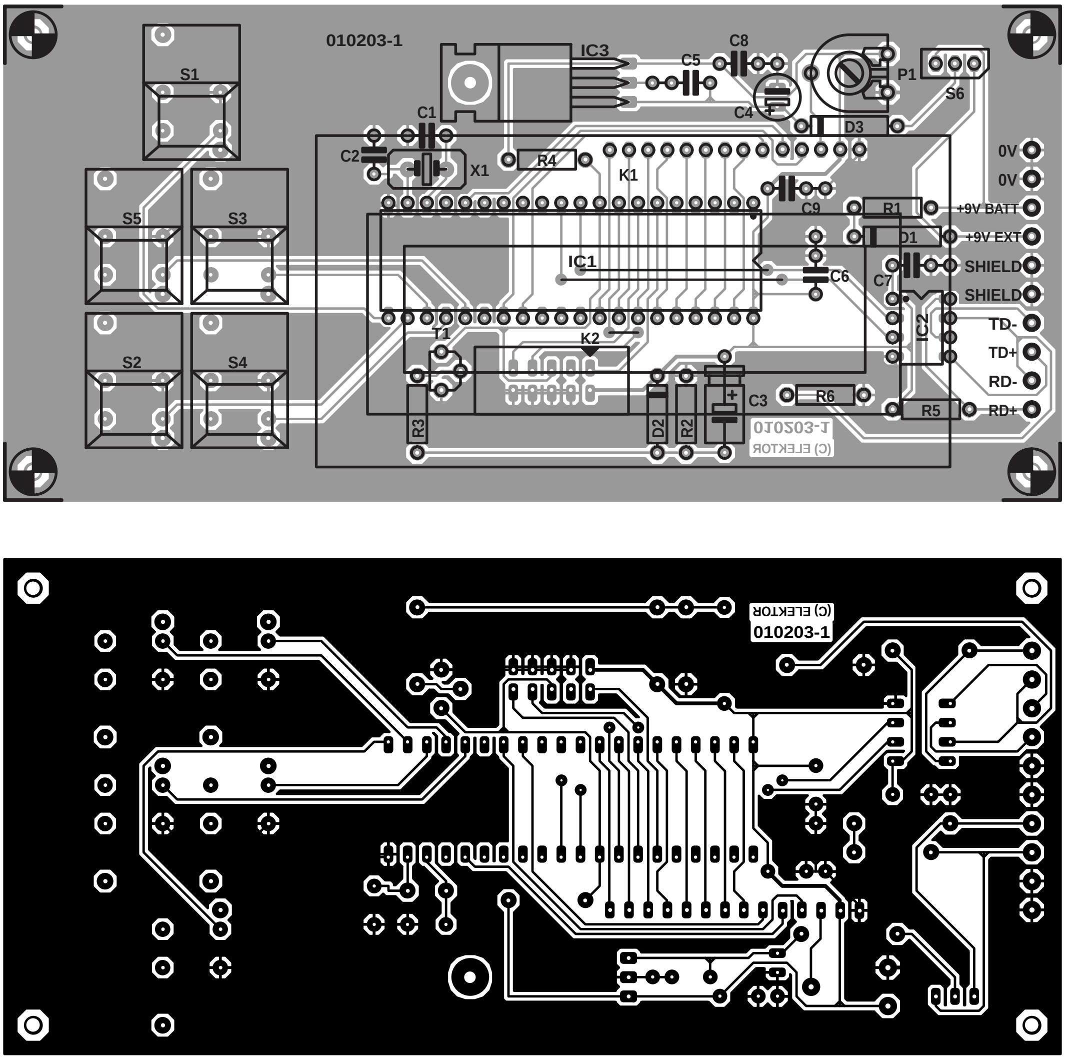


Figure 2. PCB layout and component overlay for the portable DMX tester.

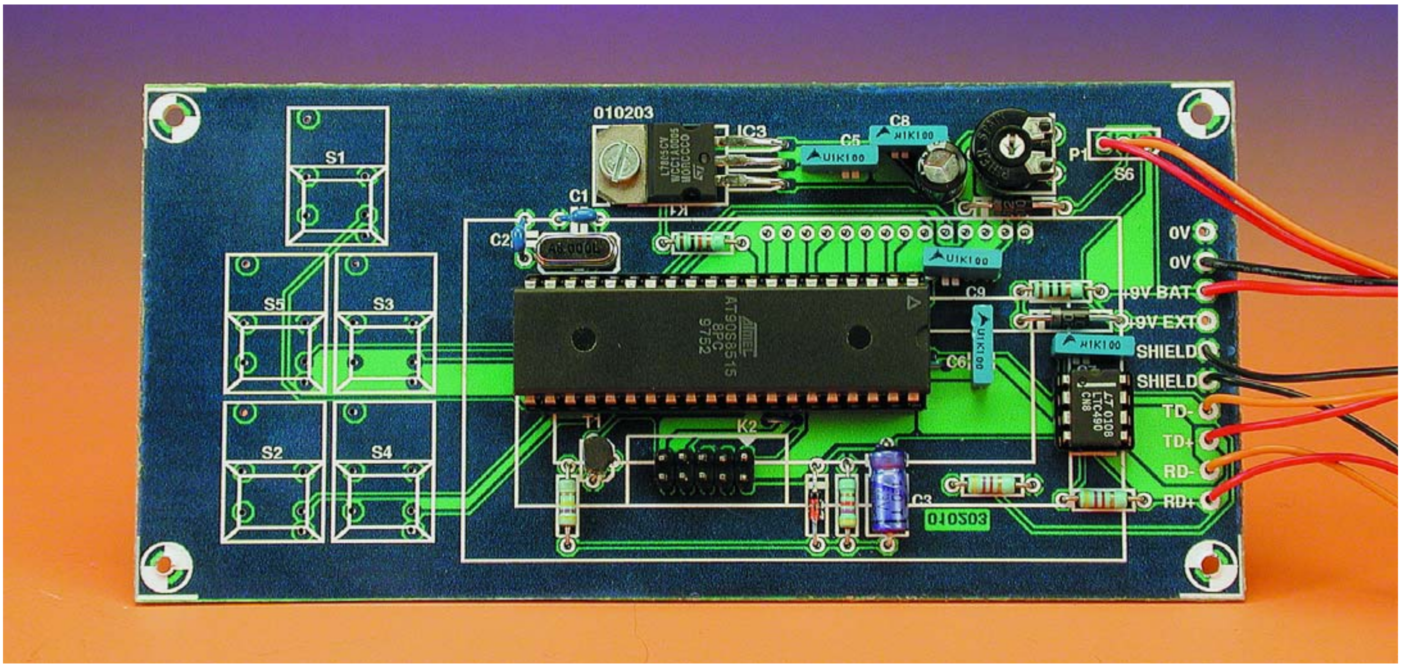


Figure 3. Completely assembled example of the circuit.

readers: an LTC490 (an equivalent of the SN75179). It takes care of the conversion from TTL to RS485. Keeping in mind the purpose of the tester, and the fact that it is battery powered, optical isolation between the interface IC and the processor was not considered necessary.

The tester can by powered via two different means: a simple mains wall adapter with approximately 9 V output or an internal NiCd battery. The selection between these two power sources is achieved simply with a three-position switch (S6). The power supply is disconnected in the centre position.

COMPONENTS LIST

Resistors:
R1 = *
R2 = 47kΩ
R3 = 470kΩ
R4 = 10kΩ
R5,R6 = 4kΩ7
P1 = 10kΩ preset

Capacitors:
C1,C2 = 33pF
C3 = 1µF 16V
C4 = 100µF 16V
C5-C9 = 100nF

Semiconductors:
D1,D3 = 1N4001
D2 = 1N4148
T1 = BC547
IC1 = AT90S8515, programmed, order code **010203-41**
IC2 = LTC490-CN8 (Linear Technology)
IC3 = 7805

Miscellaneous:
S1-S5 = pushbutton, 1 make contact
S6 = single-pole changeover switch with centre-off position
X1 = 8MHz quartz crystal
K1 = socket, 1 row of 14 contacts (+14-way header for display)
K2 = 10-way pinheader
LCD, 1 line of 16 characters, HD44780 compatible, e.g. NLC-16x1x07 (Conrad Electronics #183 261)
Optional:
5- way XLR socket, chassis mount
5-way XLR plug, chassis mount
PCB, order code **010203-1** (see Readers Services pages)
Floppy disk, source and hex-code files, order code **010203-11**

* = see text

PCB layout and disk contents also available from Free Downloads section at www.elektor-electronics.co.uk

Diode D3 protects the tester against an incorrectly connected power supply.

D1 and R1 together form a simple battery charger. This makes it possible to charge the internal battery when an external power supply is connected to the tester. The correct value for R1 depends partly on the capacity of the battery and partly on the output voltage of the mains adapter. If you are using a 9.6-V battery, then you will need to use a mains adapter that is able to provide at least 14 V, to ensure that the battery will be properly charged. The 7805 will tolerate a maximum input voltage of 18 V (this depends on the exact model), so we have plenty of margin.

The value of R1 is determined with the following formula:

$$R1 = (U_{adapter} - (1.2\ U_{battery}) - 0.7)\ /\ I_{charge}$$

In principle, assume for I_{charge} a value equal to 0.1 of the capacity of the battery, expressed in Ah.

To charge a battery rated at 9.6 V/500 mAh with a mains adapter that has an output voltage of 14 V, the value of R1 will be:

$$R1 = (14 - (1.2 \times 9.6) - 0.7)\ /0.05 = 35\ \Omega$$

(select 33 Ω from the E24 series).

The power dissipated by R1 amounts to about $33\ \Omega \times (0.05\ A)^2 = 0.1\ W$. A 0.25-W or 0.5-W resistor is therefore entirely satisfactory.

The construction

The practical realisation of a circuit built around a microcontroller, such as this one is almost of proverbial simplicity. In **Figure 2** you will find the component overlay of the matching printed circuit board.

Nonetheless, a number of aspects of the construction deserve some extra attention. A number of parts are mounted on the copper side of the PCB: the buttons and the display.

First fit all the parts at the component side, once this is done, proceed with the parts atn the copper side, starting with the pushbuttons. If you use Digitast-pushbuttons, keep in mind that these are too low in relation to the display, so they will end up recessed in the front panel. In addition, it is very difficult to make the solder connections when the pushbuttons are flush with the copper side of the PCB.

The following solution provides a means to raise the pushbuttons somewhat. First solder individual PCB pins onto the pads intended for the pushbuttons. These PCB pins are available from many electronics retailers, but if you can't find any you can use an alternative approach by using an old IC socket with turned pins. You only need to remove the pins from the plastic housing. The pushbuttons can then be inserted into the pins, and Bob's your uncle.

Once the pushbuttons are in place you get a better impression of the position of the circuit board with respect to the front panel of the enclosure. It is now easier to determine the position of the display. The mounting of the display can be done in two different ways, individual wires or with connectors. We chose the former option, since it doesn't require special parts.

We start by soldering 14 wires to the pads on the display, in such a way that they do not protrude on the display side. We then insert this assembly into the PCB and adjust the distance so that the front of the display is flush with the front of the enclosure. Everything can then be soldered to the main circuit board.

You may also decide to use a 14-way female header that fits in a corresponding male header, which is soldered to the display terminals.

Switch S3 is a single pole change-over type with centre-position. It can be mounted on the side or the front of the enclosure.

Voltage regulator IC3 is mounted horizontally to the PCB and fastened with an M3 bolt and nut. The PCB functions as the heatsink.

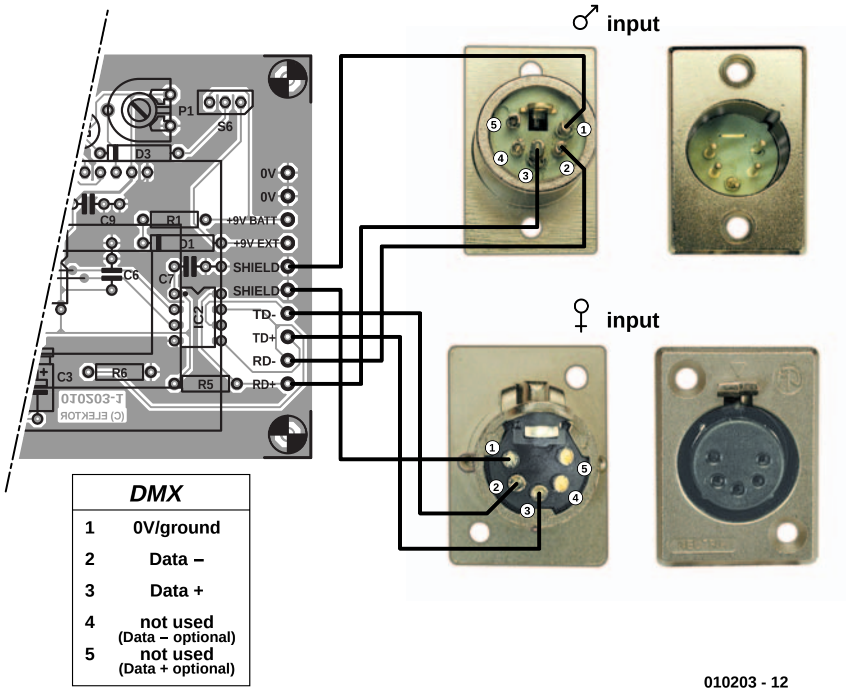
The external power supply is connected to a special low voltage connector, which is mounted directly to the PCB. Don't use a connector that can cause a short circuit when inserting or removing the plug.

When everything is finished, your circuit board should look the same as our completed prototype shown in **Figure 3**.

The dimensions of the enclosure are mainly responsible for determining the size of the 9.6-V battery (500 mAh is usually sufficient). If you do not consider portability a necessity then you can omit the entire battery altogether.

A final remark regarding the construction: the DMX connections. In principle, our tester should be provided with two XLR connectors, one for receive (male connector) and the other for transmit (female connector). Because there are two standards within the DMX world, namely XLR3 and XLR5, you will have to choose between these two connectors and make some adapter cables if necessary. It should be noted that the XLR3 types are more common than the XLR5 version — they are also cheaper.

There are two possibilities when attaching the connectors. Either mount them directly on the enclosure or on the end of a short length of flexible wire that protrudes from the enclosure. The latter solution is much more convenient (in particular between lighting fixtures). The circuit board can accommodate either method.



DMX	
1	0V/ground
2	Data –
3	Data +
4	not used (Data – optional)
5	not used (Data + optional)

Figure 4. Connection details for the XLR5-connectors and wiring diagram for the connections between the PCB and male and female XLR-chassis components.

Operation

To have a DMX tester is one thing, to know how to actually use it is another. When we are several meters up, among the lights, it would be nice to have a device that is easy to use, and in this respect the author has excelled himself.

After switching on the power, the device identifies itself with the message 'DMX TESTER Vx.y' (x.y indicates the version of the program).

The tester can function as either a transmitter or a receiver, but not both at the same time. You first have to select the operating mode. The MODE button is used for this purpose. Every time this button is pressed the operating mode is changed. The display indicates the selected mode while the button is held down.

To send data from the tester the mode '* TRANSM. MODE' is selected. To display the data on the DMX line we select 'RECEIVE MODE *'. Note that you will have to connect to the correct DMX-line corresponding to the selected operating mode.

In transmit mode, the tester can generate data on 512 DMX channels, but only the values of the first 480 are selectable. Before you can transmit a value you first have to select the channel. Channel selection is achieved with buttons CHAN+ and CHAN–. If you press and hold a button the repeat function will activate which lets you rapidly step through all the channels. The selected channel is indicated with 'C:xxx'.

It is also possible to select the DMX Start Code to be transmitted. To do this, push the channel selection buttons until the display indicates 'C:S/C' (S/C is the abbreviation for Start Code).

To change the value of a channel or the Start Code it is enough the push the DATA+ and DATA– buttons. These buttons have a repeat function as well, activated by holding the button down.

Note that the selected values are retained in memory, even of another channel is selected.

In receive mode the operation of the tester is even simpler. As soon as the tester is connected to a DMX line, the values of the first 480 channels and the Start Code are stored. The values remain in memory, even if the connection is broken,

To indicate the proper operation of the DMX line a small animation is visible on the right of the display. When the parts move in a regular pattern this indicates that the tester is receiving valid DMX frames (detection of the Mark Time). If the character on the right doesn't change then the DMX line is not working properly or the line is interrupted somewhere.

To display the values that have been received on the various channels we only have to select the channels (just as in transmit mode) with the CHAN+ and CHAN– buttons.

The receive mode has two interesting properties. The first one causes all values in memory to be completely erased when switching to receive-mode. This is in order to avoid confusion with the in transmit-mode selected values (since the same memory is used for both transmitting and receiving). If you notice that certain channels continue to have the value 0, it is possible that they have not been written to (the transmitter may be limited to 256 or 384 channels). Secondly, there is the built-in function 'BlocNote DMX'. When the tester is in receive mode, the values in the different DMX-channels are stored in memory. This memory is not erased when the tester is switched back again to transmit mode (this is opposite to the behaviour when switching to receive mode). This makes it possible, for example, to store the configuration of the DMX-channels at a particular instant and subsequently replay them at a different installation.

As you can see, this tester is bursting with features. Your mission now, is to go hunt down the problems in your own installation.

(010203-1)